IN THE CLAIMS

As shown below, please rewrite claims 22 and 29, and add new claims 37 and 38.

Sub B1

1. (Original) A data processing apparatus for executing an object-oriented operating system made up of a plurality of objects among which messages are communicated, said data processing apparatus comprising:

means for rendering an object, which has received a combining request message requesting addition of a predetermined object, to create a table data structure used for referencing to said predetermined object as a component object, and to initialize said table data structure with data of said component object, thereby constituting a composite object; and

means for creating a data structure of at least one said component object, registering the data structure in said table data structure, and registering a relationship between at least one message processing function possessed by said component object and a message interface for requesting the message processing function in the data structure of said component object.

2. (Original) A data processing apparatus according to Claim 1, wherein for initializing said table data structure, said composite object reads a data structure having predetermined settings in which a name of said component object, a number of message interfaces possessed by said component object, and a processing function for initializing said component object are described.

3. (Original) A data processing apparatus according to Claim 1, wherein said composite object has a specific execution thread and executes message processing issued to said component object with said specific execution thread.

4. (Original) A data processing apparatus according to Claim 3, wherein upon receiving a request for adding a predetermined component object, said composite object additionally registers a data structure of said predetermined component object in said data structure.

A3

5. (Original) A data processing unit according to Claim 4, further comprising means for checking, when a predetermined object is added to said composite object, whether a sequential execution relation exists between said predetermined object and all of component objects making up said composite object, and means for adding said predetermined object after the sequential execution relation has been confirmed.

6. (Original) A data processing apparatus according to Claim 5, wherein the sequential execution relation is checked by means for checking the fact that at the time when a message is transmitted to said predetermined object, all of the component objects making up said composite object are not required to run in parallel to said predetermined object, means for checking the fact that at the time when said predetermined object transmits a message to one of the component objects making up said composite object, the one component object is never already under processing of another message, and means for checking the fact that at the time when said predetermined object receives a message from one of the component objects making up said composite object, said predetermined object is never already under processing of another message.

7. (Original) A data processing apparatus according to Claim 4, wherein the apparatus checks the fact that a sequential execution relation exists between one said component object and all of the other component objects.

8. (Original) A data processing apparatus according to Claim 3, wherein upon receiving a request for separating a predetermined component object, said composite object deletes registration of a data structure of said predetermined component object from said table data structure.

A3

9. (Original) A data processing apparatus according to Claim 3, wherein said composite object includes means for checking whether a sending source of a message transmitted to said composite object is a non-component object or one said component object, and means for checking whether a destination of said message transmitted to said composite object is a non-component object or another said component object.

10. (Original) A data processing apparatus according to Claim 9, wherein when sending source of said message is one said component object and the destination of said message is another said component object, said composite object executes processing requested by said message without switching over an execution thread.

11. (Original) A data processing method for an object-oriented operating system made up of a plurality of objects among which messages are communicated, said data processing method comprising the steps of:

rendering an object, which has received a combining request message requesting addition of a predetermined object, to create a table data structure used for referencing to said predetermined object as a component object, and to initialize said table data structure with data of said component object, thereby constituting a composite object; and

creating a data structure of at least one said component object, registering the data structure in said table data structure, and registering a relationship between at least one message processing function possessed by said component object and a message interface for requesting the message processing function in the data structure of said component object.

12. (Original) A data processing method according to Claim 11, wherein for initializing said table data structure, said composite object reads a data structure having predetermined settings in which a name of said component object, a number of message interfaces possessed by said component object, and a processing function for initializing said component object are described.

13. (Original) A data processing method according to Claim 11, wherein said composite object has a specific execution thread and executes message processing issued to said component object with said specific execution thread.

14. (Original) A data processing method according to Claim 13, wherein upon receiving a request for adding a predetermined component object, said composite object additionally registers a data structure of said predetermined component object in said table data structure.

15. (Original) A data processing method according to Claim 14, further executes a step of checking, when a predetermined object is added to said composite object, whether a sequential execution relation exists between said predetermined object and all of component objects making up said composite object, and a step of adding said predetermined object after the sequential execution relation has been confirmed.

A3

16. (Original) A data processing method according to Claim 15, wherein the sequential execution relation is checked by a step of checking the fact that at the time when a message is transmitted to said predetermined object, all of the component objects making up said composite object are not required to run in parallel to said predetermined object, a step of checking the fact that at the time when said predetermined object transmits a message to one of the component objects making up said composite object, the one component object is never already under processing of another message, and a step of checking the fact that at the time when said predetermined object receives a message from one of the component objects making up said composite object, said predetermined object is never already under processing of another message.

17. (Original) A data processing method according to Claim 14, further comprising a step of checking the fact that a sequential execution relation exists between one said component object and all of the other component objects.

18. (Original) A data processing method according to Claim 13, wherein upon receiving a request for separating a predetermined component object, said composite object deletes

registration of a data structure of said predetermined component object from said table data structure.

19. (Original) A data processing method according to Claim 13, wherein said composite object includes a step of checking whether a sending source of a message transmitted to said composite object is a non-component object or one said component object, and a step of checking whether a destination of said message transmitted to said composite object is a non-component object or another said component object.

A3

20. (Original) A data processing method according to Claim 19, wherein when the sending source of said message is one said component object and the destination of said message is another said component object, said composite object executes processing requested by said message without switching over an execution thread.

21. (Original) A program providing medium for providing a data processing program for an object-oriented operating system made up of a plurality of objects among which messages are communicated, said data processing program comprising the steps of:

rendering an object, which has received a combining request message requesting addition of a predetermined object, to create a table data structure used for referencing to said predetermined object as a component object, and to initialize said table data structure with data of said component object, thereby constituting a composite object; and

creating a data structure of at least one said component object, registering the data structure in said table data structure, and registering a relationship between at least one message

processing function possessed by said component object and a message interface for requesting the message processing function in the data structure of said component object.

22. (Currently Amended) A data processing apparatus for executing an object-oriented operating system, said data processing apparatus comprising:

object constituting means for constituting objects, among which messages are communicated, by any of a composite object made up of one or more component objects and a standard object that is an object other than a composite object;

A3

identifier setting means for attaching an identifier to each of the standard object and the component objects constituted by said object constituting means so that each standard object and each component object are referenced from any object; and

execution thread control means for executing one composite object by one execution thread for the composite object included in the objects constituted by said object constituting means, said execution thread being shared by each component object of the composite object.

23. (Original) A data processing apparatus according to claim 22, further comprising added-object information reading means for, when a message for requesting a predetermined object to be added as a component object to another object is inputted, reading at least a name of the object to be added, the name being used to specify said predetermined object, and initializing method information for specifying a method in which an initializing procedure required for adding said predetermined object as a component object to said another object is described;

added-object specifying means for specifying the object to be added as a component object to said another object based on the name of the object to be added, the name being read by said added-object information reading means; and

object adding means for adding the object specified by said added-object specifying means, as a component object, to said another object by executing the method specified based on said initializing method information read by said added-object information reading means.

24. (Original) A data processing apparatus according to Claim 23, further comprising identifier creating means for, when a predetermined object is added as a component object to another object, creating a descriptor, in which information of the object to be added as a component object is stored, in correspondent relation to the identifier; and

method information storing means for storing, in the descriptor created by said identifier creating means, at least information for calling a method provided in the object to be added as a component object.

25. (Original) A data processing apparatus according to Claim 24, further comprising deleted-object information reading means for, when a message for requesting a predetermined component object to be deleted from a composite object is inputted, reading at least a name of the object to be deleted, the name being used to specify the component object to be deleted;

deleted-object specifying means for specifying the component object to be deleted from a composite object based on the name of the object to be deleted, the name being read by said deleted-object information reading means;

object deleting means for deleting the component object specified by said deleted-object specifying means from the composite object; and

descriptor deleting means for specifying, based on the identifier corresponding to the component object specified by said deleted-object specifying means, the descriptor corresponding to the specified component object and for deleting the specified descriptor.

26. (Original) A data processing apparatus according to Claim 22, wherein when constitution a composite component from a plurality of component objects, said object constituting means constitutes the composite component upon satisfying of both a condition that at the time when a message is transmitted from one component object to another one of the component objects making up the composite object including the one component object, the two component objects are not required to run in parallel, and a condition that at the time when a message is transmitted from one component object to another one of the component objects making up the composite object including the one component object, the component object on the message receiving side is never under processing of another message.

27. (Original) A data processing apparatus according to Claim 22, wherein in the case of sending a message from one object to another object, when an object on the message transmitting side and an object on the message receiving side are both component objects and said both component objects are included in the same composite object, said execution thread control means renders the object on the message receiving side to execute processing requested by the message transmitted from the object on the message transmitting side by using the same

execution thread as that used by the object on the message transmitting side without switching over the execution thread.

28. (Original) A data processing apparatus according to Claim 22, further comprising, as an application program interface used for message communication between objects, an application program interface capable of being used in common regardless of whether the objects communicating messages therebetween are each a standard object or a component object.

13

29. (Currently Amended) A data processing method executed by an object-oriented operating system, said data processing method comprising the steps of:

constituting objects, among which messages are communicated, by any of a composite object made up of one or more component objects and a standard object that is an object other than a composite object;

attaching an identifier to each of the standard object and the component objects so that each standard object and each component object are referenced from any object; and

executing one composite object by one execution thread for the composite object, said execution thread being shared by each component object of the composite object.

30. (Original) A data processing method according to Claim 29, further comprising a step of, when a message for requesting a predetermined object to be added as a component object to another object is inputted, reading at least a name of the object to be added, the name being used to specify said predetermined object, and initializing method information for specifying a method in which an initializing procedure required for adding said predetermined object as a

component object to said another object is described; a step of specifying the object to be added as a component object to said another object specified based on the name of the object to be added; and a step of adding the object specified based on the name of the object to be added, as a component object, to said another object by executing the method specified based on said initializing method information.

31. (Original) A data processing method according to Claim 30, further comprising a step of, when a predetermined object is added as a component object to another object, creating a descriptor, in which information of the object to be added as a component object is stored, in correspondent relation to the identifier; and a step of storing, in the descriptor, at least information for calling a method provided in the object to be added as a component object.

A3

32. (Original) A data processing method according to Claim 31, further comprising a step of, when a message for requesting a predetermined component object to be deleted from a composite object is inputted, reading at least a name of the object to be deleted, the name being used to specify the component object to be deleted; a step of specifying the component object to be deleted from a composite object based on the name of the object to be deleted; a step of deleting, from the composite object, the component object specified based on the name of the object to be deleted; a step of specifying, based on the identifier corresponding to the component object specified based on the name of the object to be deleted, the descriptor corresponding to the specified component object; and a step of deleting the specified descriptor.

33. (Original) A data processing method according to Claim 29, wherein when constituting a composite component form a plurality of component objects, said composite component is constituted upon satisfying of both a condition that at the time when a message is transmitted from one component object to another one of the component objects making up the composite object including the one component object, the two component objects are not required to run in parallel, and a condition that at the time when a message is transmitted from one component object to another one of the component objects making up the composite object including the one component object, the component object on the message receiving side is never under processing of another message.

A3

34. (Original) A data processing method according to Claim 29, wherein in the case of sending a message from one object to another object, when an object on the message transmitting side and an object on the message receiving side are both component objects and said both component objects are included in the same composite object, processing requested by the message transmitted from the object on the message transmitting side is executed by the object on the message receiving side by using the same execution thread as that used by the object on the message transmitting side without switching over the execution thread.

35. (Original) A data processing method according to Claim 29, wherein when communicating a message between objects, the communication is performed by using an application program interface capable of being used in common regardless of whether the objects communicating messages therebetween are each a standard object or a component object.

36. (Original) A program providing medium for providing a data processing program for an object-oriented operating system, said data processing program comprising the steps of:

constituting objects, among which messages are communicated, by any of a composite object made up of one or more objects and a standard object that is an object other than a composite object;

attaching an identifier to each of the standard object and the component objects so that each standard object and each component object are referenced from any object; and

executing one composite object by one execution thread for the composite object, said execution thread being shared by each component object of the composite object.

37. (New) A data processing apparatus according to Claim 1, wherein:

said table data structure includes a name for said component object and an object identifier for said component object; and

the data structure of said component object includes an entry table that stores the at least one registered relationships, and an oblet that includes a link table.

38. (New) A data processing method according to Claim 11, wherein:

said table data structure includes a name for said component object and an object identifier for said component object; and

the data structure of said component object includes an entry table that stores the at least one registered relationships, and an oblet that includes a link table.